# Git и GitHub.com

Одна и популярных систем контроля версий — это *git*. По своей сути — это распределенная система контроля версий (СКВ). Она является чуть ли не самой распространенной СКВ, поэтому каждый уважающий себя разработчик должен уметь ей пользоваться.

### Установка git

Для установки необходимо скачать установочной файл для конкретной операционной системы с сайта https://git-scm.com/

После установки вы получите во-первых, интерпретатор командной стоки bash, во-вторых, в системе у вас будет интегрировано приложение для контроля версий (команды git будут интегрированы в системное меню).

Поскольку git — распределенная СКВ, то существуют специальные онлайн порталы, где каждый зарегистрированный пользователь может разместить свой репозиторий. К таковым относятся bitbucket, GitHub, Gitlab и прочие. Также можно приобрести свое хранилище данных и размещать там свои репозитарии.

### Основные команды git

Начиная работать над проектом, необходимо озаботиться о том, как обеспечить сохранность текста программы от случайных изменений с помощью git.

Первым делом необходимо инициализировать репозитарий. Для этого запускаете командный интерпретатор bash, идущий в составе git для windows (для других ос просто запускаете терминал) и переходите к папке, где планируете размещать проект и вводите:

#### git init

этим вы инициализируете репозиторий (это проявляется в появлении специальной папки .git)

в случае успеха он вам напишет:

Initialized empty Git repository in path/to/directory

однако, если в этом каталоге уже есть папка .git выдаст вам

Reinitialized existing Git repository in path/to/directory

теперь вы можете посмотреть что же находится в git репозитории выполнив команду:

#### git status

в результате вы увидите нечто такое:

```
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    app.2.cpp
    arrays
    arrays.cpp
    hanoi.cpp
    test
    test.cpp
```

nothing added to commit but untracked files present (use "git add" to track)

Все верно. Ваш репозиторий пуст. Теперь вы должны поместить отслеживаемые объекты в git, чтобы она мониторила их изменения. Делается это следующей командой:

**git add** <имя файла>

Пусть то будет, например, файл arrays.cpp

git add arrays.cpp

Теперь вызов команды статуса покажет нам следующее

```
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file: arrays.cpp
Untracked files:
  (use "git add <file>..." to include in what will be committed)
  .DS_Store
  app.2.cpp
  arrays
  hanoi.cpp
  test
  test
  test.cpp
```

Мы добавили в git arrays.cpp, но изменения не зафиксированы, между тем git теперь будет следить за изменением этого файла

Чтобы зафиксировать изменения, нужно ввести:

#### git commit

После этого, git запустит редактор операционной системы по-умолчанию со следующим текстом:

*# Please enter the commit message for your changes. Lines starting* 

```
# with '#' will be ignored, and an empty message aborts the commit.
#
# On branch master
# Changes to be committed:
   new file: arrays.cpp
#
#
# Untracked files:
   .DS_Store
#
#
  app.2.cpp
#
   arrays
#
  hanoi.cpp
#
   test
#
   test.cpp
#
```

где вы должны добавить в самом начале строки, которые описывают сделанные вами изменения в репозитории. Если вы ничего не введете и закроете редактор, git напишет вам:

Aborting commit due to empty commit message.

Если изменения были внесены, то вы получите что-то воде этого:

```
[master 05f54ea] qweqwe
1 file changed, 141 insertions(+)
create mode 100644 arrays.cpp
```

Все, теперь вы зафиксировали изменения в вашем репозитории. Так как пока в нем один файл, то зафиксированы изменения этого файла.

Теперь, если изменить файл arrays.cpp, а затем спросить статус., мы получим:

```
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
  modified: arrays.cpp
Untracked files:
  (use "git add <file>..." to include in what will be committed)
  .DS_Store
  app.2.cpp
  arrays
  hanoi.cpp
  test
  test.cpp
no changes added to commit (use "git add" and/or "git commit -a")
```

То есть, git обнаружил, что файл arrays.cpp изменился. Чтобы зафиксировать его новое состояние нужно снова выполнить

# git add arrays.cpp git commit

## Перейдем к Github

Если возникает необходимость опубликовать наш репозиторий для совместного доступа, можно воспользоваться сервисом GitHub.

После регистрации, мы получаем возможность создавать свои репозитории онлайн или выгружать локальные репозитории.

Очень хорошее руководство по git и его командам приведено здесь.

From: http://lidarbackup.dvo.ru/dokuwiki/ - Записки репетитора Permanent link: http://lidarbackup.dvo.ru/dokuwiki/doku.php/posts:vcs\_git\_github Last update: 2022/04/27 09:07